



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTER DE L'ENSEIGNEMENT SUPERIEUR ET DE LA
RECHERCHE SCIENTIFIQUE



Université Echahid Hamma Lakhdar El OUED

Mémoire

En vu de l'obtention du diplôme de **Licence en Informatique**
Option : **systemes informatiques**

Thème

Conception et implémentation d'une application mobile pour:
food ordering and deliverery

Réalisé par :

- ❖ Mr.Harrane Hatem Mouaouia
- ❖ Mr.Mennani Baha Eddine
- ❖ Mr.Sebouai Hicham

Encadré par :

- ❖ Dr.Beggas Mounir

Promotion 2020/2021

TABLE DES MATIÈRES

1	Le développement des applications mobiles	4
1.1	Introduction	4
1.2	Définition d'application mobile	5
1.3	Les Types d'application mobile	6
1.3.1	Application native :	6
1.3.2	Application Web Mobile :	7
1.3.3	Comparaison entre application native et Web App :	8
1.4	Android	9
1.5	Smartphones et systèmes d'exploitation pour téléphones Android	9
1.5.1	Présentation du smartphone :	9
1.5.2	Systèmes d'exploitation pour téléphones Android :	10
2	Analyse et conception	13
2.1	Introduction	13
2.2	Problématique	13
2.3	Méthode de solution	13
2.4	Avantages de commander de la nourriture en ligne	14
2.5	La modélisation UML	15
2.6	Diagramme de classe	15
2.7	Diagramme de use case	16
2.8	Diagramme de séquences	17
3	Implementation	20
3.1	Introduction	20
3.2	Environnement logiciel	20
3.3	Flutter [4]	20
3.4	Dart [5]	20
3.5	Visual Studio Code	21
3.6	Firebase	22
3.7	L'environnement de test	22
3.8	Interfaces applicatives	23
3.8.1	L'application client :	23
3.8.2	L'application d'administration de restaurant	34

TABLE DES FIGURES

1.1	Applications mobiles	5
1.2	Application mobile native Guide Alger	6
1.3	Page d'accueil du site web mobile de l'université de Stanford	7
1.4	La différence entre une application Web et une application mobile	8
1.5	Informations sur le téléphone	10
1.6	Systèmes d'exploitation mobiles	11
2.1	Diagramme de classe générale	15
2.2	Diagramme de use case générale	16
2.3	Diagramme de séquences d'inscrire un client	17
2.4	Diagramme de séquences d'authentification	17
2.5	Diagramme de séquences d'envoyer une commande	18
2.6	Diagramme de séquences d'ajouter des plats est voir les commandes	18
3.1	Interface de l'environnement Visual Studio 1.56.2	21
3.2	interface de démarrage	23
3.3	Connexion client	24
3.4	code de connexion avec e-mail et mot de passe	24
3.5	Utilisateurs enregistrés de l'application	25
3.6	Créer un nouveau compte	25
3.7	code créer un utilisateur avec Email un mot de passe	26
3.8	Données personnelles des clients enregistrés dans Firestore Database	26
3.9	Voir et choisir le restaurant	27
3.10	Tiroir pour afficher des informations et des tâches	28
3.11	code de déconnexion	28
3.12	Modification des informations personnelles du client	29
3.13	code de modifier les informations	29
3.14	commande et la sélection des plats	30
3.15	méthode créer une commande	31
3.16	Liste des commandes et de leur contenu enregistrés dans Firebase (RealTime DATABASE)	31
3.17	Entrez le lieu et l'heure	32
3.18	La commande a été envoyée	33
3.19	interface de démarrage	34
3.20	Connexion client	35
3.21	Créer un nouveau compte	36

3.22	Données personnelles des administrateurs des restaurants enregistrés dans Firestore Database	36
3.23	Ajouter un plat et le prix	37
3.24	Liste des plats enregistrés avec leur prix par le directeur du restaurant sur Firebase (Realtime DATABASE)	37
3.25	Code pour ajouter un plat à la base de données Firebase(RealTime DATABASE)	38
3.26	alert pour confirme l'ajoute de plat	39
3.27	Tiroir pour afficher des informations et des tâches	40
3.28	Modification des informations personnelles du client	41
3.29	le menu de plat	42
3.30	Code de méthode pour obtenir le menu des plats du firebase	42
3.31	Code pour suppression d'un plat dans Firebase, avec affichant une alerte contenant un message confirmant la suppression.	43
3.32	les Commandes de clients	43
3.33	Code de méthode qui récupère la liste des commandes du firebase	44
3.34	Contenu de la commande	45
3.35	Code pour supprimer la commande sur le Firebase lorsque la commande est prête	45

Rèsumé

La terrible propagation des restaurants et l'augmentation notable de ceux-ci, ont entraîné le problème de pression et d'incompréhension (retard, perte de temps....) c'est pourquoi les services de livraison de plat sont apparus, et donc cette application agit comme un intermédiaire entre le client et le restaurant, car il choisit le plat et le restaurant selon son envie et de sa place.

Abstract

The terrible spread of restaurants and the noticeable increase in them, resulted in the problem of pressure and misunderstanding (delay, loss of time ...) that is why food delivery services appeared, and therefore this application acts as an intermediary between the customer and the restaurant, as he chooses the food and the restaurant according to his desire and from his place.

ملخص

إن الإنتشار الرهيب للمطاعم والتزايد الملحوظ عليها ، نتج عنه مشكلة الضغط وسوء الفهم (تأخير ، خسارة وقت) لهذا ظهرت خدمات توصيل الطعام ، ولذلك فإن هذا التطبيق يعمل كوسيط بين الزبون والمطعم ، كما يختار الأكلات والمطعم على رغبته ومن مكانه .

Introduction générale

INTRODUCTION GÉNÉRALE

Dans un monde actif et continuellement évolutif, la motivation d'avoir des moyens performants et efficaces de communication et d'échange d'informations devient de plus en plus fondamentale. Cette motivation donne naissance à une révolution favorisant le travail à distance et l'accès aux besoins en temps réduit à l'aide d'internet qui a bouleversé les habitudes de travail dans de nombreux métiers. D'après beaucoup d'analyses et statistiques effectuées, il s'avère que de plus en plus d'internautes se connectent désormais à internet via leurs téléphones portables. Nous remarquons ces dernières années un développement exponentiel des appareils mobiles qui sont répandus comme une traînée de poudre dans le monde en développant et révolutionnant le domaine des communications notamment dans les zones rurales. Dans ce cadre, les Smartphones apparaissent pour rompre avec nos anciennes idées sur les téléphones portables et donner une autre dimension à cette technologie tout en intégrant de nouveaux apports à la téléphonie mobile et en attirant la clientèle grâce à l'ergonomie exponentielle et révolutionnaire. Grâce à l'arrivée de ce miracle de la technologie, les usages des téléphones mobiles vont être modifiés d'une manière significative. Les appareils mobiles joueront le rôle de lien entre le monde virtuel du web et le monde physique qui nous entoure. Ils fournissent aux utilisateurs individuels et aux communautés un accès précieux à toute une série de services d'information à des fins personnelles et commerciales.

C'est dans cette optique que s'inscrit notre travail, qui consiste à développer une application Android pour la gestion d'un service de restauration. Pour l'organisation de notre travail, nous avons adopté la structure suivante :

Le premier chapitre intitulé « Le développement des applications mobiles », donnera un aperçu général sur les mobiles.

Le second chapitre intitulé « Analyse et conception », sera consacrée à l'analyse et la conception de notre application, en utilisant le langage UML.

Le troisième chapitre, intitulé « Perception », présente les différents outils utilisés Pour développer notre application, suivi de quelques captures d'écran

illustrant cela Traiter.

Enfin, une conclusion générale dans laquelle sont récapitulés les points clés de notre travail, les difficultés rencontrées et les améliorations.

*Le développement des
applications mobiles*

CHAPITRE 1

LE DÉVELOPPEMENT DES APPLICATIONS MOBILES

1.1 Introduction

L'importance des applications électroniques dans nos vies augmente avec le temps car elles fournissent des services qui facilitent les fonctions de la vie des utilisateurs.

Et par les applications, nous entendons celles sur les magasins électroniques qui offrent un package dans un domaine spécifique pour l'utilisateur.

Ces dernières années, les applications se sont considérablement développées et avec le développement des téléphones portables et des systèmes d'exploitation, l'importance de ces applications s'est accrue.

Nous avons remarqué que les téléphones intelligents sont terriblement répandus dans divers domaines et domaines, ce que nous pouvons résumer certains de ses objectifs comme la réalisation de profits croissants au fil du temps, le maintien du nombre d'utilisateurs et la tentative de dépasser les concurrents qui fournissent le même service dans le même domaine.

Avec le développement de ces applications, cela pourrait provoquer une renaissance révolutionnaire de la technologie, d'autant plus que les utilisateurs de téléphones intelligents sont également à la hausse.

Les téléphones portables ont créé de merveilleuses opportunités de partage et de connexion.

Dans cette optique, le projet de formation à la fin de l'étude vise à concevoir et mettre en œuvre une application de transformateur pour la commande et la livraison de nourriture.

1.2 Définition d'application mobile

Les applications mobiles sont des logiciels développés pour les petits appareils portables, comme les téléphones mobiles, Smartphones, PDA et ainsi de suite. Les applications mobiles peuvent être chargées par l'utilisateur sur l'appareil mobile et téléchargées sur les plateformes de téléchargement d'applications (Apps Stores) ou sur Internet.

Ils existent plusieurs systèmes d'exploitation mobiles les plus populaires sont : Windows Mobile, Android, iOS, Symbian, Java ME et Palm [1].



FIG. 1.1: Applications mobiles

1.3 Les Types d'application mobile

1.3.1 Application native :

Une application native est une application dédiée à un périphérique mobile (Smartphone, Tablette,... etc). Elle est installée directement sur l'appareil et accessible à travers une icône. Les utilisateurs acquièrent généralement ces applications à travers une boutique en ligne ou sur un store en ligne comme l'App Store ou Android Apps sur Google Play. Une application native est développée pour qu'elle fonctionne sur une plateforme précise, elle peut profiter de toutes les fonctionnalités qui existent dans l'appareil : GPS, Accéléromètre, Appareil photo, la liste des Contacts... Les applications mobiles peuvent aussi interpréter les gestes tactiles sur l'écran de l'appareil, peuvent utiliser le système de notification de l'appareil et peuvent même fonctionner hors ligne.



FIG. 1.2: Application mobile native Guide Alger

1.3.2 Application Web Mobile :

Une application web mobile (ou webapp) est une application utilisée via Internet et possédant des fonctionnalités spécifiques pour les appareils mobiles. Elles ne sont pas réellement des applications mais plutôt des sites web adaptés à l'usage mobile. Elles sont accessibles exactement comme les sites web c'est-à-dire via le navigateur Web de l'appareil mobile et n'ont pas besoin d'être téléchargées et installées sur l'appareil pour fonctionner, mais leur installation reste possible et permet de faciliter l'accès à une application par la création de l'icône sur l'écran d'accueil de l'appareil.

Les applications Web Mobile ne bénéficient pas de toutes les fonctionnalités de l'appareil mobile. Pour prendre avantage de ces fonctions il faut opter pour une application native ou une solution hybride que nous allons entamer après la comparaison entre site mobile et application mobile.

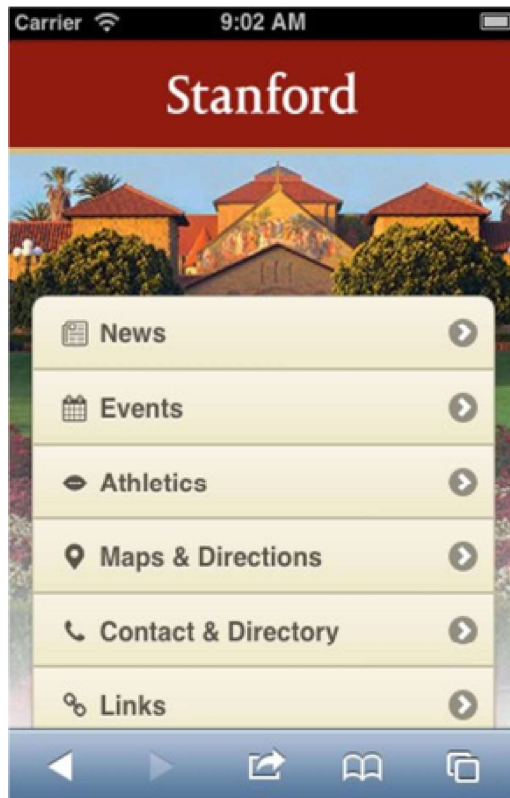


FIG. 1.3: Page d'accueil du site web mobile de l'université de Stanford

1.3.3 Comparaison entre application native et Web App :

	Web app	Application mobile
 Compatibilité multiplateforme	<ul style="list-style-type: none"> + Compatibilité avec tous les terminaux - Les fonctionnalités disponibles dépendent du navigateur utilisé 	<ul style="list-style-type: none"> - Une application par plateforme (iOS, Android), même si le code peut en partie être mutualisé grâce à un langage cross-plateforme
 Ergonomie et graphisme	<ul style="list-style-type: none"> - Le graphisme est le même pour tous les OS - Ergonomie limitée : le rendu graphique est dépendant du navigateur 	<ul style="list-style-type: none"> + Sur-mesure et donc adaptés aux usages mobile de chaque OS
 Fluidité et performances	<ul style="list-style-type: none"> - Dépend de la qualité du réseau et du navigateur utilisé. Possibilité de temps de chargement plus longs, lags... 	<ul style="list-style-type: none"> + Puissance, qualité et fluidité optimales
 Fonctionnalités	<ul style="list-style-type: none"> - N'accède pas à l'ensemble des fonctionnalités native du téléphone - Expérience utilisateur inégale en fonction de l'OS et du navigateur 	<ul style="list-style-type: none"> + Accès à 100% des fonctionnalités natives du téléphone
 Distribution	<ul style="list-style-type: none"> + Accessible dès la mise en ligne 	<ul style="list-style-type: none"> - Doit être soumises au stores - 24 à 72h de temps de validation
 Monétisation	<ul style="list-style-type: none"> - Impossible de vendre le téléchargement de l'app + Vente en ligne + Monétisation par la publicité 	<ul style="list-style-type: none"> - Commission des stores + Possibilité de faire payer le téléchargement + Achat intégré à l'application via le store (achat in-app) + Monétisation par la publicité
 Référencement	<ul style="list-style-type: none"> + SEO : optimisations possibles pour se référencer sur les moteurs de recherche 	<ul style="list-style-type: none"> - ASO : optimisation limitées sur le store + SEO : possibilité de créer une landing page sur son site pour renvoyer vers le store
 Mise à jour du contenu	<ul style="list-style-type: none"> + Simple et rapide 	<ul style="list-style-type: none"> + Possible via un back-office - Nécessite une re-soumission sans back office
 Évolution et maintenance	<ul style="list-style-type: none"> + Peu coûteuse 	<ul style="list-style-type: none"> + À réaliser pour chaque OS (sauf s'il s'agit d'une application cross-plateforme) - Re-soumission obligatoire

FIGURE 1.4 – La différence entre une application Web et une application mobile

1.4 Android

Android est un système d'exploitation mobile fondé sur le noyau Linux et développé par Google.

Lancé en juin 2007 à la suite du rachat par Google en 2005 de la startup du même nom, le système avait d'abord été conçu pour les smartphones et tablettes tactiles, puis s'est diversifié dans les objets connectés et ordinateurs comme les télévisions (Android TV), les voitures (Android Auto), les Chromebook (Chrome OS qui utilise les applications Android) et les smartwatch (Wear OS).

En 2015, Android est le système d'exploitation mobile le plus utilisé dans le monde, devant iOS d'Apple, avec plus de 80% de parts de marché dans les smartphones pour l'ensemble de ses versions et adaptations. Android est en effet diffusé sous trois formes :

- Il peut être modifié par les constructeurs qui y ajoutent leurs surcouches, apportant ainsi des fonctionnalités supplémentaires mais au détriment du délai d'obtention des nouvelles mises à jour qui est parfois important.

- Il peut également être installé sans surcouche comme sur les smartphones Android One qui bénéficient de ce fait rapidement des nouvelles versions du système ou encore sur les smartphones Android Go, qui disposent quant à eux d'une version allégée capable de fonctionner de manière rapide sur du matériel d'entrée de gamme.

- Android existe enfin aussi sous la forme de différentes versions alternatives également appelées ROM Custom (ou forks en anglais) notamment /e/ (anciennement Eelo).

1.5 Smartphones et systèmes d'exploitation pour téléphones Android

1.5.1 Présentation du smartphone :

Le smartphone ou « téléphone intelligent » désigne un téléphone mobile doté de fonctionnalités évoluées qui s'apparentent à celles d'un ordinateur : navigation sur Internet, lecture de vidéos, de musique, jeux vidéo, courrier électronique, vidéoconférence, bureautique légère.

Muni d'un processeur puissant, souvent multicœur, il embarque une série de capteurs (boussole, accéléromètre, gyroscope, GPS) qui lui permettent de faire fonctionner des applications dédiées à l'activité physique, de navigation assistée ainsi que des jeux que l'on peut contrôler d'un simple mouvement. Les smartphones sont généralement dotés d'un appareil photo-vidéo et d'une

caméra frontale dont les performances ne cessent de progresser[2].



FIGURE 1.5 – Informations sur le téléphone

1.5.2 Systèmes d'exploitation pour téléphones Android :

Les systèmes d'exploitation mobiles (OS) peuvent être définis comme les logiciels permettant à un smartphone ou un téléphone mobile basique de fonctionner. Ils permettent de ce fait aux utilisateurs de pouvoir passer un appel téléphonique, naviguer sur leurs téléphones parmi toutes les rubriques, télécharger des applications ou encore paramétrer et personnaliser leurs smartphones.

Il existe une vingtaine de systèmes d'exploitation sur le marché, dont certains sont développés directement par et pour les fabricants de téléphones eux-mêmes, d'autres forment des conglomérats pour mutualiser les coûts et créer des synergies, et certains veulent juste se concentrer sur le développement de systèmes d'exploitation.

Depuis maintenant de nombreuses années, Android, développé par les ingénieurs de Google, s'impose comme l'OS de référence du marché, loin devant ses concurrents. Il atteint ainsi en 2014 le milliard de smartphones équipés, quand iOS à la seconde place ne dépasse pas les 180 millions. L'avantage de ce système d'exploitation se résume dans le fait qu'il est mis à disposition de tous les fabricants de téléphones, à condition qu'ils respectent certaines conditions liées à la mise en avant de services Google. Pour le reste, l'OS est personnalisable à

souhait, par les consommateurs comme par les fabricants. Comme le démontre le graphique ci-dessous, de nos jours, la majorité des téléphones vendus dans le monde tournent sous Android. On peut ainsi citer de très nombreuses marques de smartphones, parmi lesquelles Samsung, Acer, HTC, LG, Sony ou encore Huawei[3].

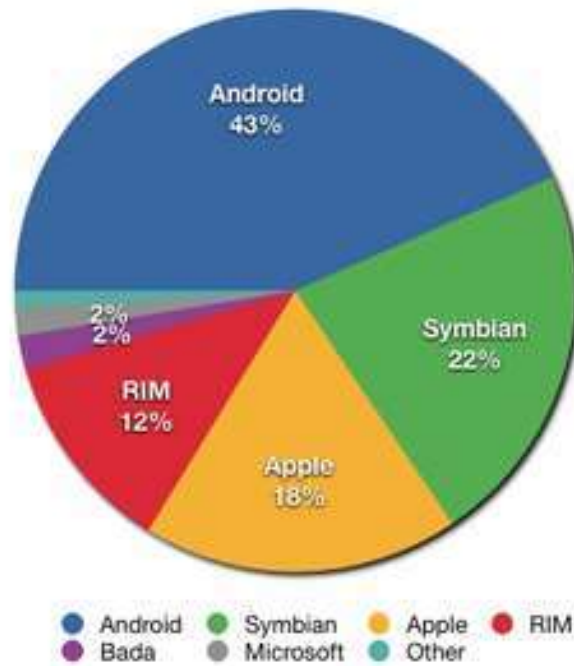


FIGURE 1.6 – Systèmes d'exploitation mobiles

Analyse et conception

CHAPITRE 2

ANALYSE ET CONCEPTION

2.1 Introduction

Le but de ce chapitre est de faire une bonne réalisation de notre système ou bien de notre projet, alors il faut faire une bonne modélisation pour garantir la qualité et la rentabilité. Donc pour garantir tout ça, il faut utiliser une méthode ou démarche de conception.

2.2 Problématique

De nos jours, les gens sont tellement occupés qu'ils n'ont pas le temps d'aller au restaurant et d'acheter leur plat préféré, et n'ont peut-être pas le temps de quitter leur domicile ou leur bureau pour acheter. La nourriture, la distance ou même les appels téléphoniques peuvent être une raison pour laquelle les gens sont incapables de répondre à leurs besoins.

Le principal problème avec la commande téléphonique est qu'elle crée un certain nombre de malentendus, la plupart du temps la personne recevant votre demande n'entend pas clairement votre voix en raison d'une interruption de la file d'attente ou de toute autre raison entraînant la livraison de la mauvaise demande. Et le client peut prendre beaucoup de temps pour réfléchir à ce qu'il veut.

2.3 Méthode de solution

Un moyen facile de sortir de ce problème est d'utiliser des services de livraison de nourriture en ligne, et fournir de nombreux services de commande de nourriture en ligne. En fait, la restauration rapide est sur la liste des aliments les plus commandés au monde et selon les experts. la plupart des fast-foods ne tardent pas à être préparés, mais il existe de nombreux avantages à commander en ligne.

- Facilité d'accès : Avec un service de livraison de nourriture en ligne, la première fonctionnalité dont nous pouvons profiter est de choisir parmi toute une gamme d'aliments, et nous pouvons consulter leurs menus en ligne pour voir quels types d'aliments ils servent.
- Explorer de nouveaux endroits : est l'un des principaux avantages de passer votre commande de nourriture en ligne, par conséquent, vous pouvez choisir parmi de nombreuses options concernant la nourriture et les restaurants.
En fait, les menus en ligne sont explicites, ce qui permet aux utilisateurs de passer plus facilement leurs commandes, de cette façon nous pouvons vérifier les aliments proposés par différents restaurants sans vous rendre dans chaque restaurant un par un, et cela peut nous faire gagner beaucoup de temps et d'efforts.
- Économies de coûts : Les services de restauration en ligne peuvent nous faire économiser beaucoup de temps et d'argent, la raison en est que vous n'aurez pas à vous asseoir dans votre voiture et à aller au restaurant, donc acheter de l'essence nous coûte de l'argent et beaucoup de temps et d'efforts pour voyager.
C'est donc une bonne idée de commander notre nourriture en ligne et d'utiliser notre service de livraison de nourriture en ligne, nous pouvons passer notre commande dans le confort de notre foyer tant que nous sommes connectés à Internet.

2.4 Avantages de commander de la nourriture en ligne

À l'ère d'Internet, notre présence en ligne est plus qu'une exigence, surtout si notre activité est liée aux aliments qui peuvent être commandés en ligne, il y a un certain nombre d'avantages pour nous.

- Rapide et facile : L'un des principaux avantages de la commande de nourriture dans les restaurants en ligne est qu'elle permet aux clients de commander plus facilement et plus rapidement de la nourriture et d'atteindre les restaurants. Il est également idéal pour les clients et les restaurants. Nous devons donc passer à cette technique pour en tirer plus d'avantages.
- Il n'y a aucun risque de malentendu : En commandant de la nourriture en ligne, nous pouvons clairement la personnaliser en fonction des besoins, ce qui réduit la confusion et garantit la satisfaction du client.

- Moins de tracas : Il faut beaucoup de temps au client pour réfléchir à ce qu'il veut vraiment, mais avec l'aide de la commande de nourriture en ligne, ce tracas est éliminé et on ne peut commander que ce qu'il veut, ce qui fait gagner du temps.

2.5 La modélisation UML

Le Langage de Modélisation Unifié, de l'anglais Unified Modeling Language (UML), est un langage de modélisation graphique à base de pictogrammes conçu comme une méthode normalisée de visualisation dans les domaines du développement logiciel et en conception orientée objet.

2.6 Diagramme de classe

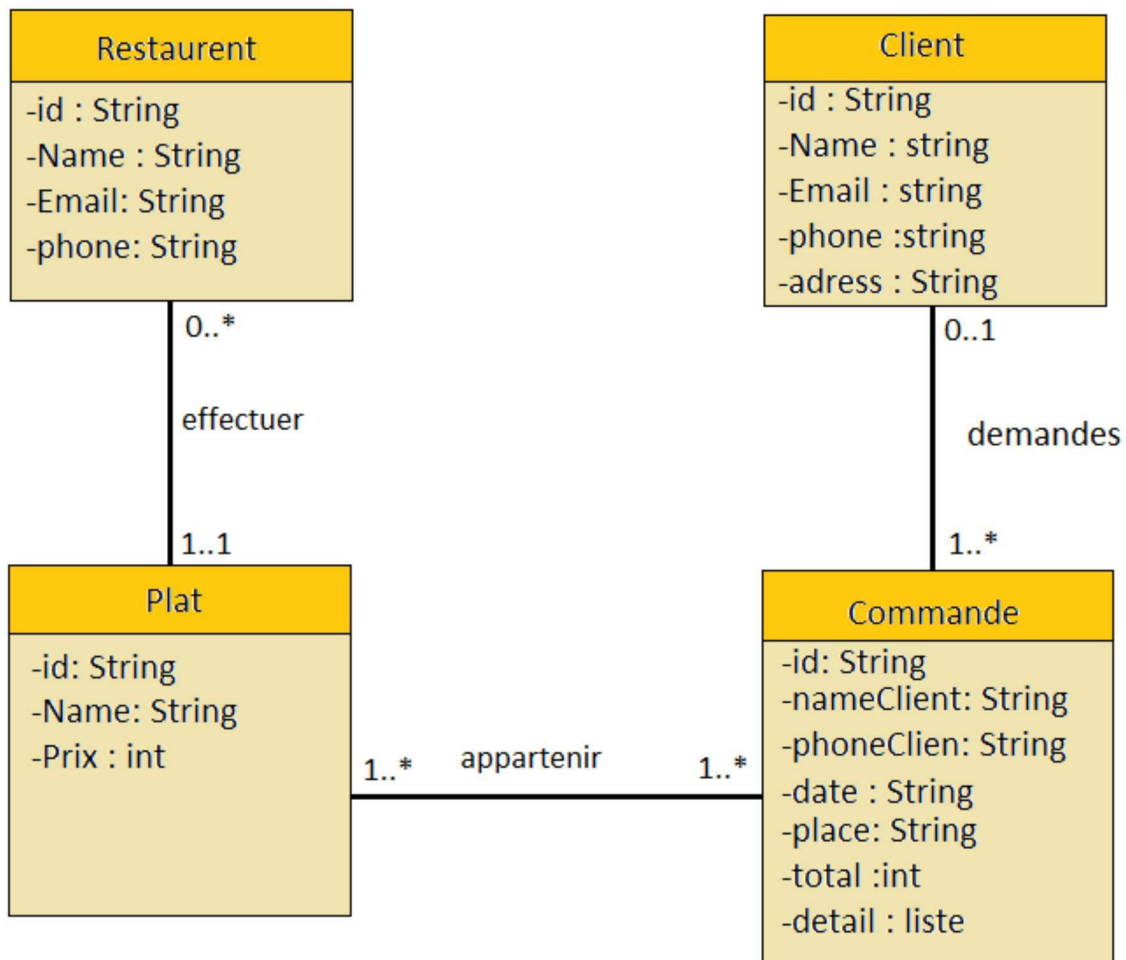


FIGURE 2.1 – Diagramme de classe générale

2.7 Diagramme de use case

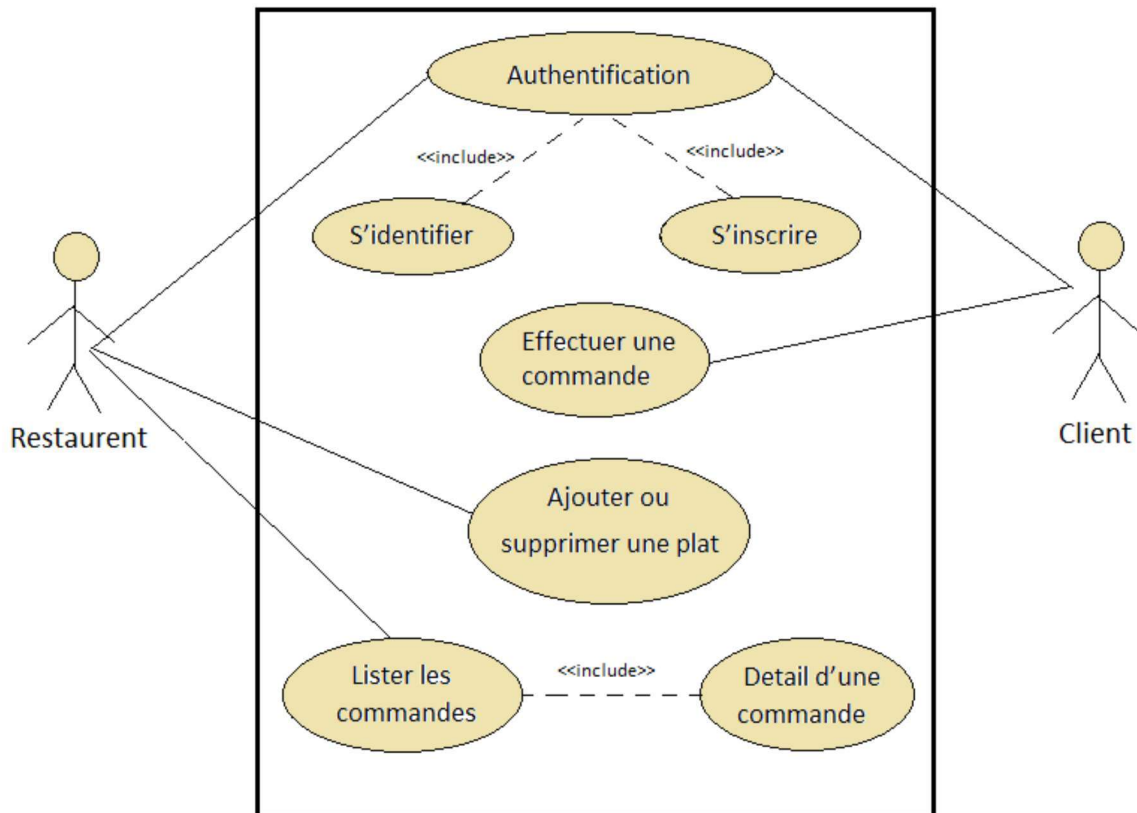


FIGURE 2.2 – Diagramme de use case générale

2.8 Diagramme de séquences

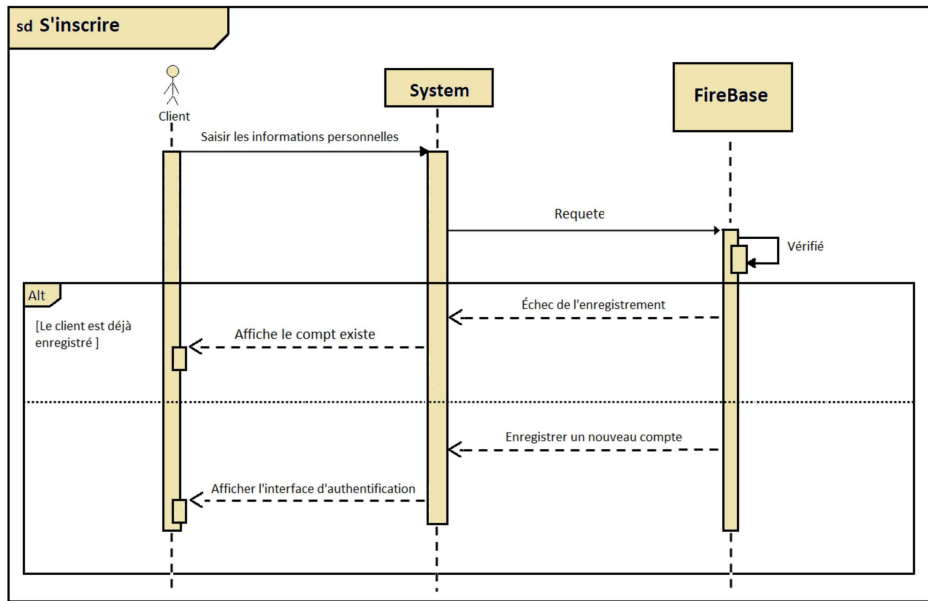


FIGURE 2.3 – Diagramme de séquences d'inscrire un client

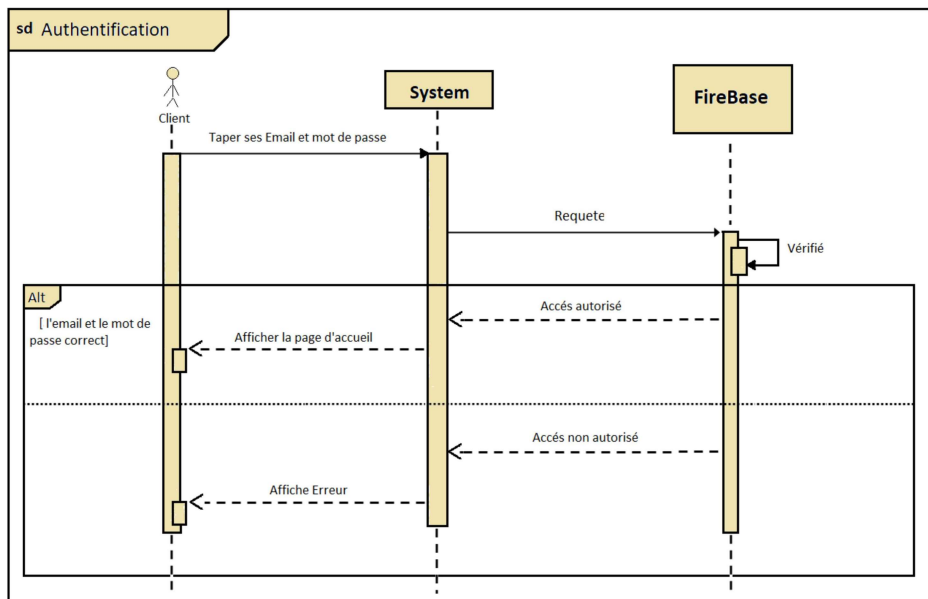


FIGURE 2.4 – Diagramme de séquences d'authentification

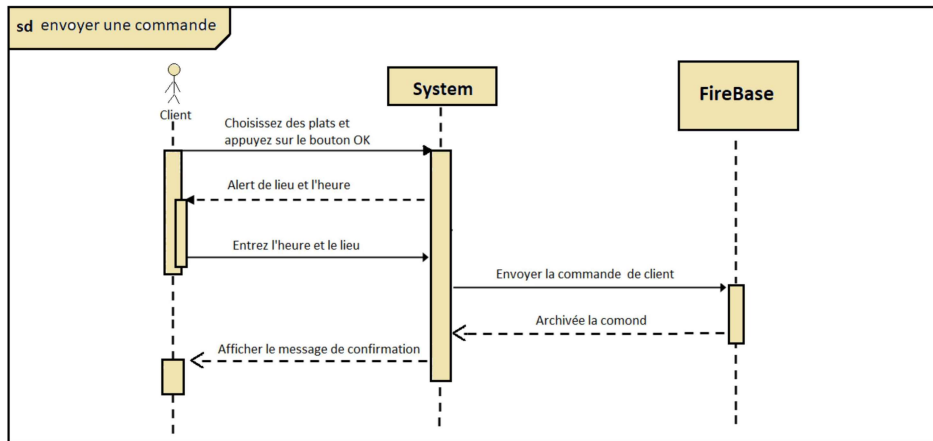


FIGURE 2.5 – Diagramme de séquences d'envoyer une commande

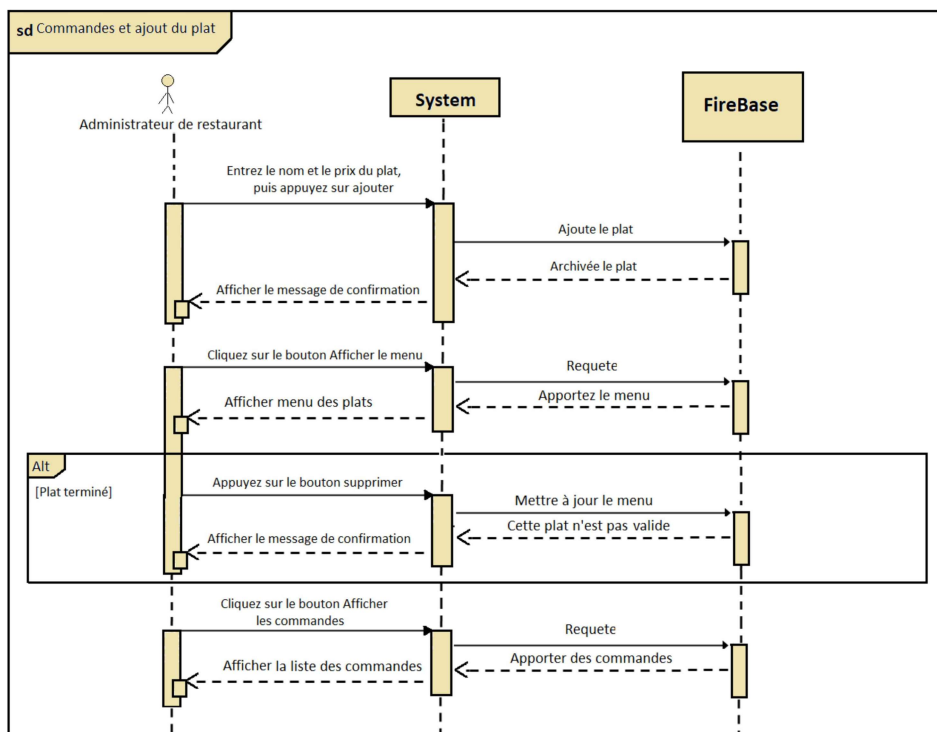


FIGURE 2.6 – Diagramme de séquences d'ajouter des plats est voir les commandes

Implementation

CHAPITRE 3

IMPLEMENTATION

3.1 Introduction

Après avoir compléter l'étude conceptuelle dans le chapitre précédent, nous allons aborder la partie implémentation dans ce qui suit. Nous commençons par présenter l'environnement matériel et logiciel, ensuite nous présenterons les différentes interfaces de l'application.

3.2 Environnement logiciel

Pour pouvoir exécuter la commande dans de bonnes conditions, il est bien entendu indispensable de choisir l'environnement de travail en fonction de vos besoins. Nous avons choisi de réaliser une application mobile dans le système Android qui nous oblige à travailler sous flutter avec le langage Dart et le SDK Android pour la développer, afin d'avoir un fichier apk qui sera installé plus tard sur différents types de terminaux mobiles fonctionnant Android 6.0 ou supérieur.

3.3 Flutter [4]

Flutter est un kit de développement de logiciel (SDK) d'interface utilisateur open-source créé par Google. Il est utilisé pour développer des applications pour Android, iOS, Linux, Mac, Windows, Google Fuchsia et le web à partir d'une seule base de code.

3.4 Dart [5]

Dart est un langage de programmation optimisé pour les applications sur plusieurs plateformes. Il est développé par Google et est utilisé pour créer des applications mobiles, de bureau, de serveur et web.

3.5 Visual Studio Code

Visual Studio Code est un éditeur de code extensible développé par Microsoft pour Windows, Linux et macOS[6].

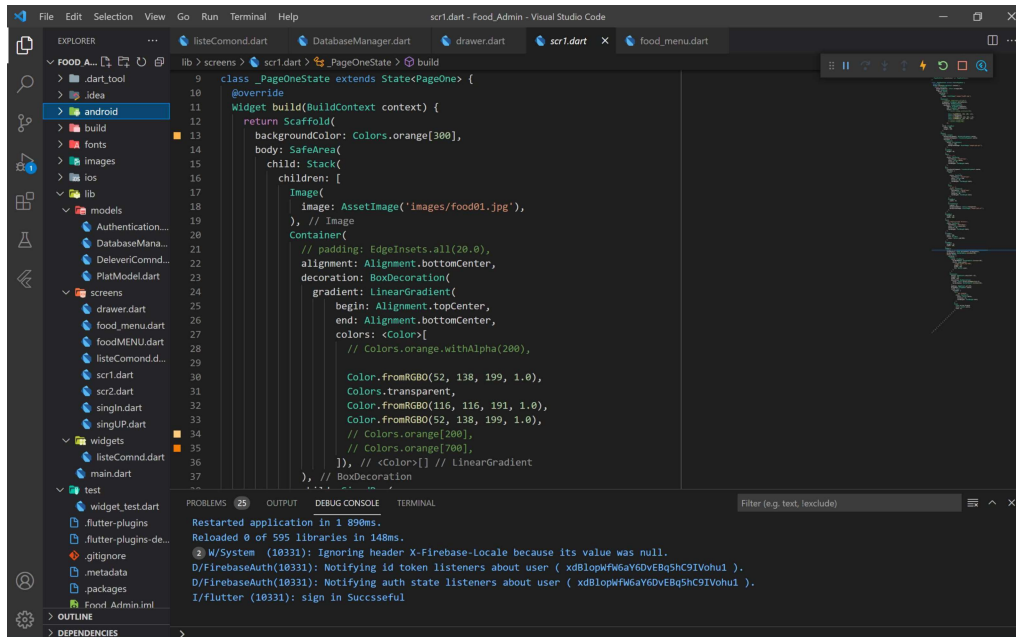


FIGURE 3.1 – Interface de l'environnement Visual Studio 1.56.2

3.6 Firebase

Firebase est un ensemble de services d'hébergement pour n'importe quel type d'application (Android, iOS, Javascript, Node.js, Java, Unity, PHP, C++ ...). Il propose d'héberger en NoSQL et en temps réel des bases de données, du contenu, de l'authentification sociale (Google, Facebook, Twitter et Github), et des notifications, ou encore des services, tel que par exemple un serveur de communication temps réel.

Les applications qui utilisent Firebase intègrent une bibliothèque qui permet les diverses interactions possibles[7].

Ce sont les bibliothèques que nous avons utilisées pour créer notre application.

- Firebase auth : 0.15.5+3
- Firebase database : 3.1.3
- Cloud firestore : 0.13.4
- Firebase core : 0.4.4+3

3.7 L'environnement de test

Nous avons utilisé l'émulateur Pixel 4 XL API 28 .

3.8 Interfaces applicatives

Dans ce qui suit nous présenterons les différentes interfaces de l'application en citant les détails de chaque interface .

3.8.1 L'application client :

Interface de démarrage de l'application :

Cette interface apparaît lorsque vous ouvrez l'application.



FIGURE 3.2 – interface de démarrage

interface de connexion :

Dans cette interface, l'application se connecte en entrant l'email et le mot de passe (si enregistré) et s'il n'est pas enregistré le client ira enregistrer un nouveau compte, et lorsqu'il appuie sur le bouton "SING IN", les données du client (email et le mot de passe) vont à Firebase pour entrer dans l'application.

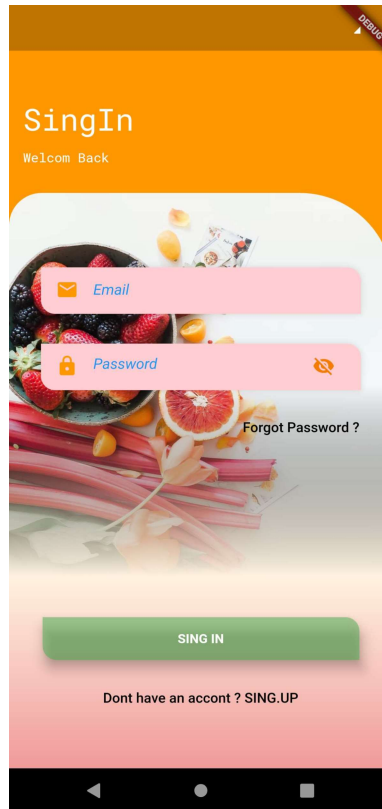


FIGURE 3.3 – Connexion client

```
final FirebaseAuth _auth = FirebaseAuth.instance;

Future loginUser(String email, String password) async {
  try {
    AuthResult result = await _auth.signInWithEmailAndPassword(
      email: email, password: password);
    FirebaseUser user = result.user;
    return user;
  } catch (e) {
    print(e.toString());
  }
}
```

FIGURE 3.4 – code de connexion avec e-mail et mot de passe

Identifiant	Fournisseurs	Date de création	Dernière connexion	UID utilisateur ↑
burak@yahoo.com	✉	26 mai 2021	9 juin 2021	2N3m2COYRAReszc5alsCvf2t6v2
baha@yahoo.com	✉	26 mai 2021	6 juin 2021	adQ5XFwieWU24NBGkryZ3HywW...
hichem@yahoo.com	✉	6 juin 2021	6 juin 2021	IJCxVGSGBUM7avxk9FydrZglHyc2
hatem@yahoo.com	✉	26 mai 2021	8 juin 2021	xdBlopWfW6aY6DvEBq5hC9IVohu1

FIGURE 3.5 – Utilisateurs enregistrés de l’application

Interface pour créer un nouveau compte pour se connecter à l’application :

Dans cette interface, un nouveau client est enregistré en saisissant ces informations et en appuyant sur le bouton "SING UP" ces informations sont enregistrées au niveau de Firebase.

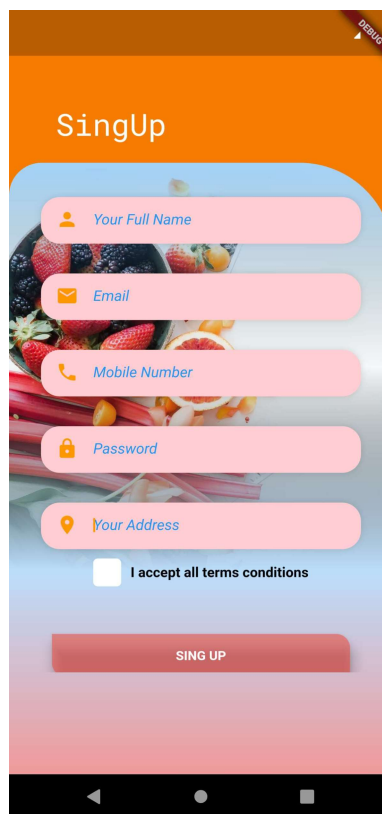


FIGURE 3.6 – Créer un nouveau compte

```

final FirebaseAuth _auth = FirebaseAuth.instance;

Future createNewUser(String name, String email, String phone, String password,
String address) async {
  try {
    AuthResult result = await _auth.createUserWithEmailAndPassword(
      email: email, password: password);
    FirebaseUser user = result.user;
    await DatabaseManager()
      .creatUserData(name, email, phone, address, user.uid);
    return user;
  } catch (e) {
    print(e.toString());
  }
}

```

FIGURE 3.7 – code créer un utilisateur avec Email un mot de passe

The screenshot shows the Firebase console interface. On the left, a sidebar shows the project structure with 'Client' selected. The main area displays a document with the following fields:

4eKS2q3wugfB1FojUAdDWC71dWm1	
adQ5XFWieWU24NBGkryZ3HywWc12	
iJCxVG5GBUM7avxk9FydrZgIHyc2	
xdB1opWfW6aY6DvEBq5hC9IVohu1	

On the right, the document's content is displayed as a form:

- Email: "hatem@yahoo.com"
- Name: "Harrane Hatem Mouaouia"
- Phone: "+213659360145"
- adress: "19 Mars oude"

FIGURE 3.8 – Données personnelles des clients enregistrés dans Firestore Database

Interface pour afficher les restaurants existants :

Dans cette interface, le restaurant est sélectionné

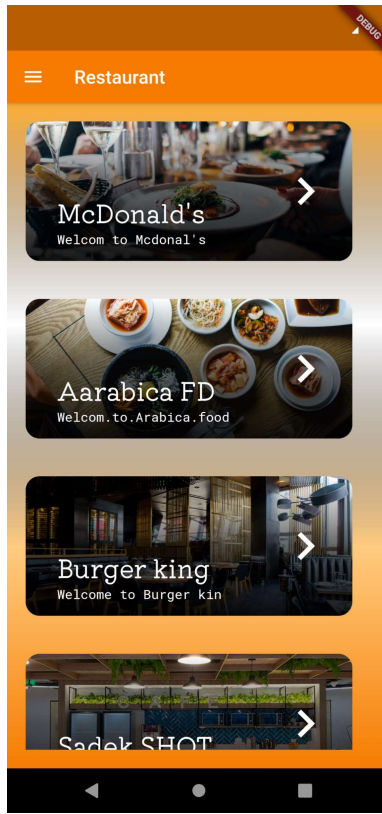


FIGURE 3.9 – Voir et choisir le restaurant

Tiroir affichant quelques informations et fonctions :

Lorsque vous cliquez sur la barre d'état système, l'email et l'identifiant du client apparaissent ainsi que le bouton de modification des données et le bouton de déconnexion de l'application

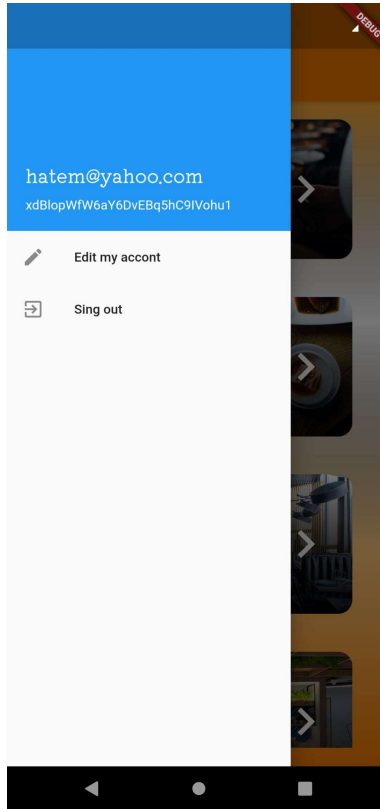


FIGURE 3.10 – Tiroir pour afficher des informations et des tâches

```
final FirebaseAuth _auth = FirebaseAuth.instance;

Future singOut() async {
  try {
    return _auth.signOut();
  } catch (error) {
    print(error.toString());
    return null;
  }
}
```

FIGURE 3.11 – code de déconnexion

Interface d'édition d'informations :

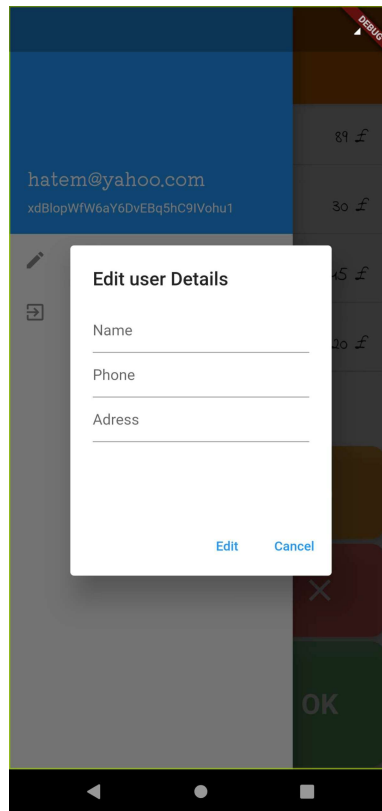


FIGURE 3.12 – Modification des informations personnelles du client

```
final CollectionReference profileList =
    Firestore.instance.collection('Client');
Future updateUserList(
    String name, String phone, String adress, String uid) async {
    return await profileList
        .document(uid)
        .updateData({'Name': name, 'Phone': phone, 'adress': adress});
}
```

FIGURE 3.13 – code de modifier les informations

L'interface principale pour la commande et la sélection des plats :

Sur cette page, les plats sont sélectionnés à partir du menu affiché et répertorié par le directeur du restaurant, indiquant le prix total du menu sélectionné

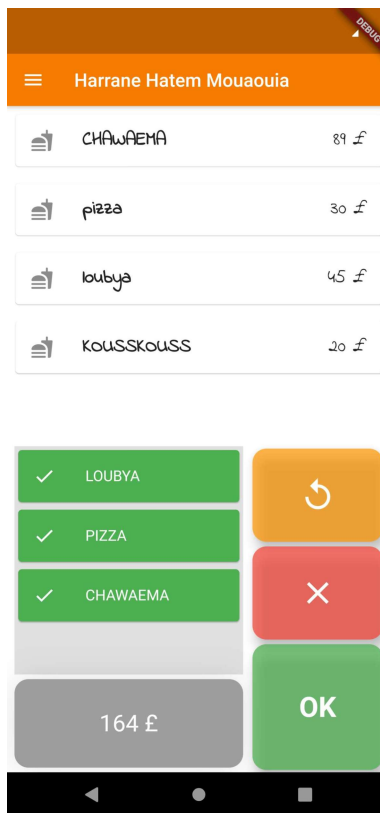


FIGURE 3.14 – commande et la sélection des plats

```

Future creatComodData(int total, String pphone, String nname, String date,
String place, String uid, List<OnCommd> items) async {
  await FirebaseApp.configure(
    name: 'food delivery',
    options: const FirebaseOptions(
      googleAppID: '1:1083587334075:android:e899dc623e8f32a5bb7e9e',
      apiKey: 'AIzaSyD_sh05mf09lhy2TVWhfo1VUmARK1G4suk',
      databaseURL: 'https://food-delivery-d15ae-default-rtdb.firebaseio.com/'
    ),
  ).then((value) {
    app = value;
    final DatabaseReference db = FirebaseDatabase(app: app).reference();

    DatabaseReference keyRef = db.reference().child('Commd').push();
    keyRef
      .set({
        'total': total,
        'phoneClient': pphone,
        'nameClient': nname,
        'date': date,
        'place': place,
        'uid': uid
      })
      .then((value) => {
        items.forEach((element) async {
          final DatabaseReference db =
            FirebaseDatabase(app: app).reference();
          db
            .reference()
            .child('Commd')
            .child(keyRef.key)
            .child("detail")
            .push()
            .set({'name': element.nomplat});
        });
      });
    return null;
  });
}

```

FIGURE 3.15 – méthode créer une commande

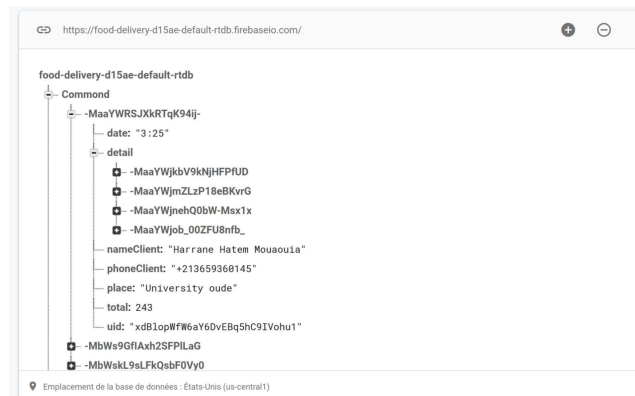


FIGURE 3.16 – Liste des commandes et de leur contenu enregistrés dans Firebase (RealTime DATABASE)

Dans cette interface, une alerte s'affiche pour renseigner l'heure et le lieu nécessaires à la réception de la commande

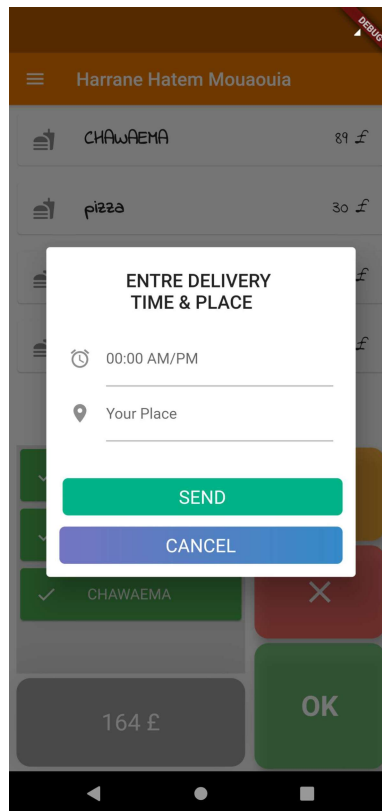


FIGURE 3.17 – Entrez le lieu et l'heure

Dans cette interface, le message d’alerte de confirmation de commande est affiché

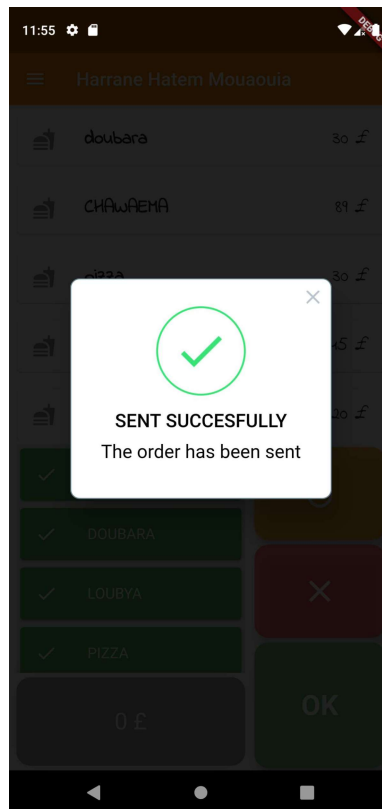


FIGURE 3.18 – La commande a été envoyée

3.8.2 L'application d'administration de restaurant

Interface de démarrage de l'application :

Cette interface apparaît lorsque vous ouvrez l'application

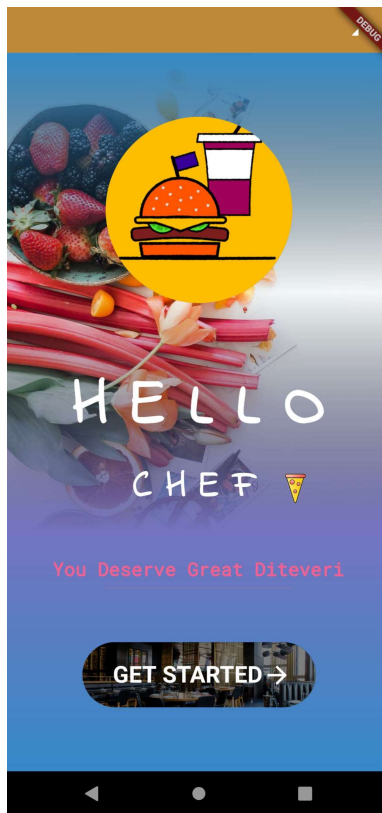


FIGURE 3.19 – interface de démarrage

interface de connexion :

Cette interface fonctionne de la même manière que la figure 3.3

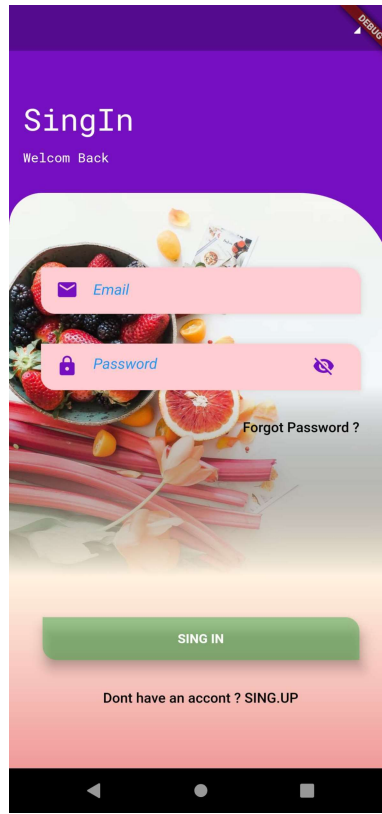


FIGURE 3.20 – Connexion client

Interface pour créer un nouveau compte pour se connecter à l'application :

Cette interface fonctionne de la même manière que la figure 3.5

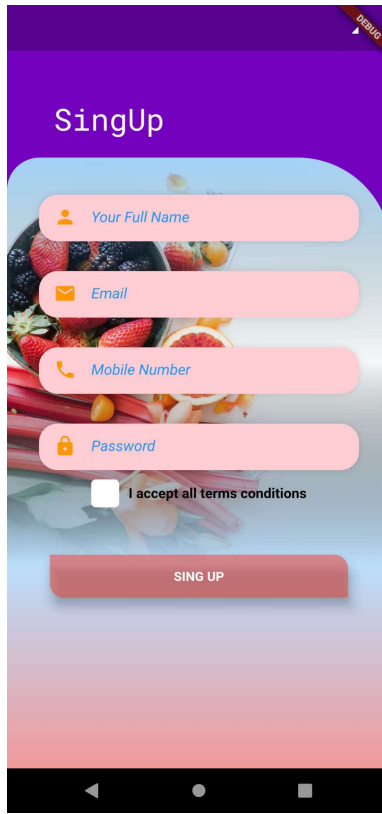


FIGURE 3.21 – Créer un nouveau compte

Admine > 2N3m2COYRAR...		
food-delivery-d15ae	Admine	2N3m2COYRARReszcr5alsCvf2t6v2
+ Commencer une collection	+ Ajouter un document	+ Commencer une collection
Admine >	2N3m2COYRARReszcr5alsCvf2t6v2	+ Ajouter un champ
Client	4bw4nMaL9LhrtEYQAtUdD4SI0D12	Email: "burak@yahoo.com"
		Name: "burakk"
		Phone: "0987654321"

FIGURE 3.22 – Données personnelles des administrateurs des restaurants enregistrés dans Firestore Database

Dans cette interface, un plat est ajouté au menu et le prix du plat est entré

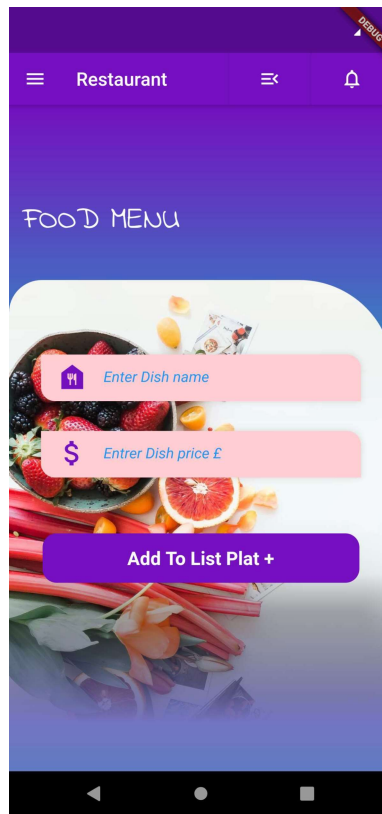


FIGURE 3.23 – Ajouter un plat et le prix

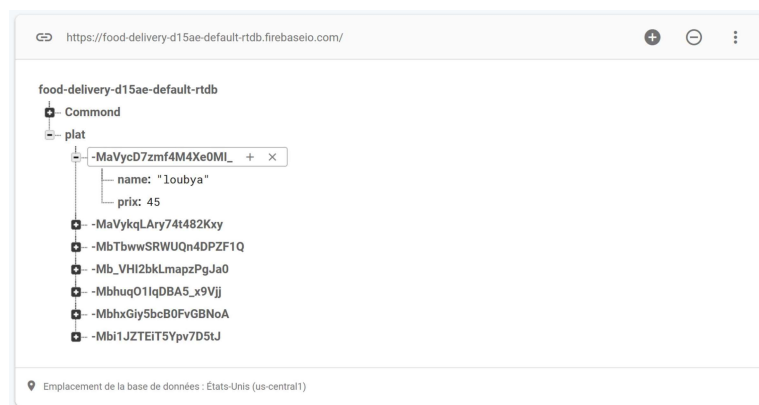


FIGURE 3.24 – Liste des plats enregistrés avec leur prix par le directeur du restaurant sur FireBase (Realtime DATABASE)

```
final FirebaseApp app =
    await FirebaseApp.configure(
        name: 'food-delivery',
        options:
            const FirebaseOptions(
                googleAppID:
                    '1:1083587334075:android:e899dc623e8f32a5bb7e9e',
                apiKey:
                    'AIzaSyD_sh05mf091hy2TVWhfo1VUmARKlG4suk',
                databaseURL:
                    'https://food-delivery-d15ae-default-rtdb.firebaseio.com/',
            ), // FirebaseOptions
    ).then((value) {
        final DatabaseReference db =
            FirebaseDatabase(
                app: value) // FirebaseDatabase
                .reference();

        db
            .reference()
            .child('plat')
            .push()
            .set({
                'name':
                    _nomplatController
                    .text,
                'prix': int.parse(
                    _prixController
                    .text)
            })
            .then((value) => {
                _nomplatController
                    .text = "",
                _prixController
                    .text = ""
            });
    });
```

FIGURE 3.25 – Code pour ajouter un plat à la base de données Firebase(RealTime DATABASE)

Interface pour confirmer l'ajout de plat au menu

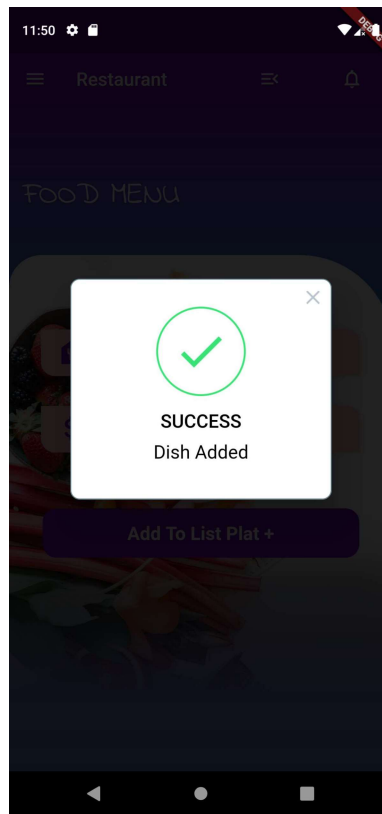


FIGURE 3.26 – alert pour confirme l'ajoute de plat

Tiroir affichant quelques informations et fonctions :

Cette interface fonctionne de la même manière que la figure 3.8

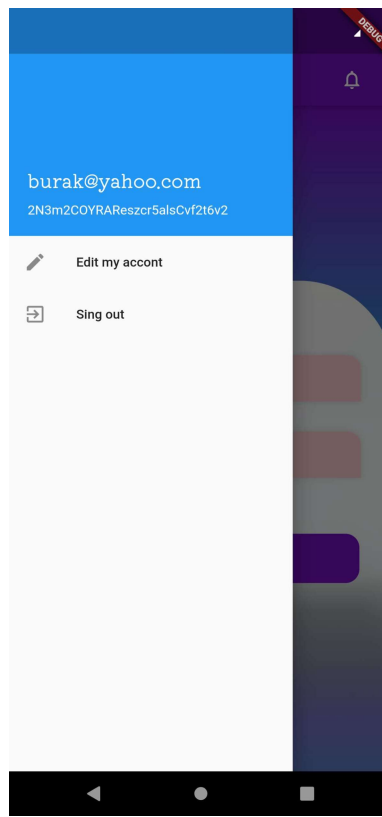


FIGURE 3.27 – Tiroir pour afficher des informations et des tâches

Interface d'édition d'informations :

Cette interface fonctionne de la même manière que la figure 3.10

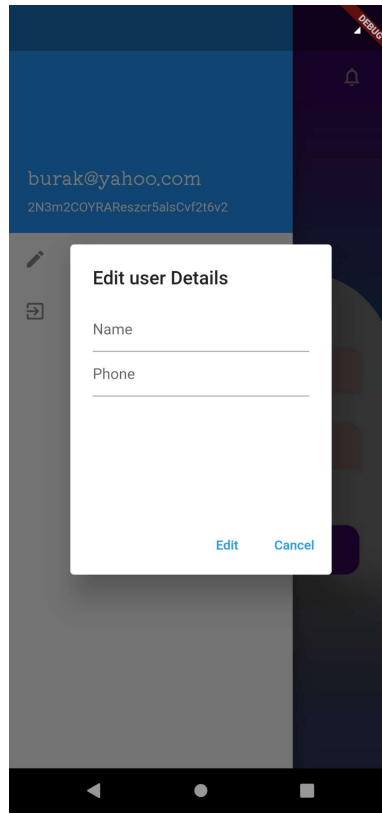


FIGURE 3.28 – Modification des informations personnelles du client

L'interface affiche le menu :

Dans cette interface, la liste des plats répertoriés par l'administrateur s'affiche avec la possibilité de supprimer le plat lorsqu'il est épuisé.

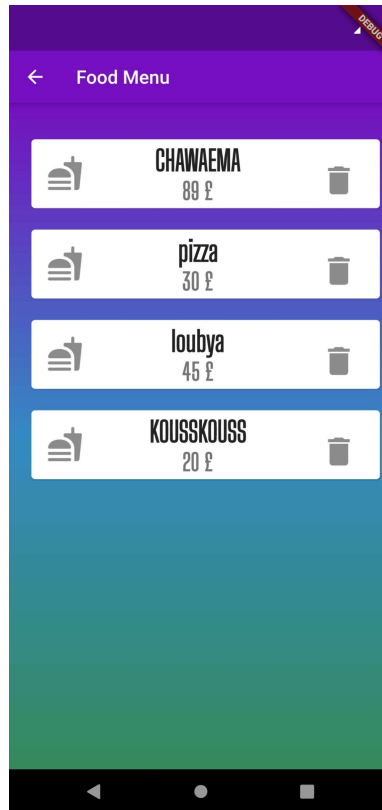


FIGURE 3.29 – le menu de plat

```

Map<dynamic, dynamic> dataplat = Map();
fetchDatabaseListplat() async {
  final FirebaseApp app = await FirebaseApp.configure(
    name: 'food-delivery',
    options: const FirebaseOptions(
      googleAppID: '1:1083587334075:android:e899dc623e8f32a5bb7e9e',
      apiKey: 'AIzaSyD_sh0Smf091hy2TWWhfo1VUmARK1G4suk',
      databaseURL: 'https://food-delivery-d15ae-default-rtdb.firebaseio.com/'
    ),
  ),
).then((value) {
  final DatabaseReference db = FirebaseDatabase(app: value).reference();
  db.reference().child("plat").once().then((DataSnapshot dataSnapshot) => {
    dataplat = dataSnapshot.value as Map;
    dataplat.forEach((key, value) {
      loaded = true;
      setState(() {
        userProfilesListplat
          .add(PlatModel(key, value["name"], value["prix"]));
      });
    });
  });
  return null;
});
}

```

FIGURE 3.30 – Code de méthode pour obtenir le menu des plats du firebase

```

onPressed: () {
  setState(() {
    FirebaseDatabase.instance
      .reference()
      .child('plat')
      .child(userProfilesListplat[index].id)
      .remove();

    userProfilesListplat.removeAt(index);
  });
  Alert(
    context: context,
    type: AlertType.success,
    title: "DELETE",
    desc: "is succeeded",
    buttons: [],
  ).show(); // Alert
},

```

FIGURE 3.31 – Code pour suppression d’un plat dans Firebase, avec affichant une alerte contenant un message confirmant la suppression.

Interface affichant les commandes envoyées par le client :

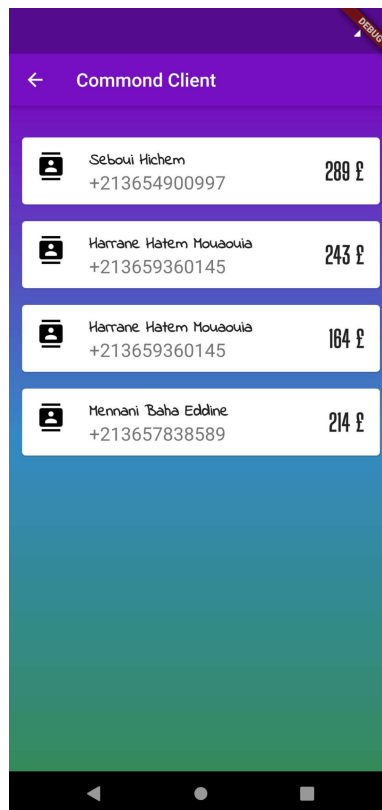


FIGURE 3.32 – les Commandes de clients

```
fetchDatabaseList() async {
  final FirebaseApp app = await FirebaseApp.configure(
    name: 'food-delivery',
    options: const FirebaseOptions(
      googleAppID: '1:1083587334075:android:380a8062ef60df3ebb7e9e',
      apiKey: 'AIzaSyD_sh05mf091hy2TVWhFo1VUmARK1G4suk',
      databaseURL: 'https://food-delivery-d15ae-default-rtdb.firebaseio.com/',
    ),
  ).then((value) {
    final DatabaseReference db = FirebaseDatabase(app: value).reference();
    db
      .reference()
      .child("Command")
      .once()
      .then((DataSnapshot dataSnapshot) => {
        data = dataSnapshot.value as Map,
        data.forEach((key, value) {
          loaded = true;
          setState(() {
            Map<dynamic, dynamic> detailMap = value["detail"] as Map;
            List<String> listdetail = List();
            detailMap.forEach((key, value) {
              listdetail.add(value["name"]);
            });
            userProfilesList.add(DeleverComnd(
              key: key,
              date: value["date"],
              name: value["nameClient"],
              phone: value['phoneClient'],
              place: value['place'],
              total: value["total"],
              detail: listdetail,
              uid: value["uid"]));
          });
        });
      });
    return null;
  });
});
```

FIGURE 3.33 – Code de méthode qui récupère la liste des commandes du firebase

Dans cette interface, le contenu des informations de la commande (menu des plats, heure et lieu de la commande) est affiché avec la possibilité de supprimer la commande lorsqu'elle est préparée

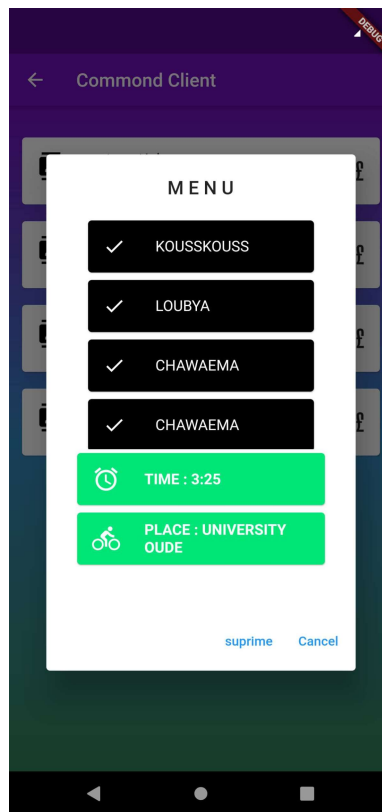


FIGURE 3.34 – Contenu de la commande

```
onPressed: () {
  setState(() {
    FirebaseDatabase.instance
      .reference()
      .child('Command')
      .child(userProfilesList[
        index]
        .key)
      .remove()
      .then(() {
        Navigator.of(context)
          .pop();
      });

    userProfilesList
      .removeAt(index);
  });
  child: Text('suprime'),
], // FlatButton
```

FIGURE 3.35 – Code pour supprimer la commande sur le Firebase lorsque la commande est prête

Conclusion générale

CONCLUSION GÉNÉRALE

Ce projet nous a permis d'avoir une idée plus claire et plus précise sur le développement sous Android ; ce système d'exploitation qui a envahit le marché des terminaux mobiles.

Durant notre étude nous avons vu de façon détaillée le système, son architecture, ses dernières versions ainsi que les outils utilisés afin de réaliser ce projet, d'ailleurs nous avons opté pour l'utilisation des dernières technologies d'où notre travail s'est reposé sur une implémentation sous Flutter avec le langage Dart et une base de données Firebase.

Pour arriver à un résultat nous avons suivie une démarche de modélisation en commençant par la spécification du cas d'utilisation dans un premier temps, suivi d'une étude de conception précise et détaillée.

Enfin il y a une la mise en œuvre de notre application, l'obtention des résultats et l'objectif est atteint, comme l'amélioration de notre travail reste envisageable et comme perspective.

Bibliographie

BIBLIOGRAPHIE

- [1] Site Internet de : <http://mobiledevices.about.com/od/glossary/g/What-Is-A-Mobile-Application.htm> , [visité le 07/06/2021]

- [2] Site Internet de : <https://www.futura-sciences.com/tech/definitions/smartphone-smartphone-1954/> , [visité le 05/06/2021]

- [3] Site Internet de : <https://www.monpetitmobile.com/choisir-mobile/systemes-exploitation-smartphones> , [visité le 08/06/2021]

- [4] Site Internet de : « Flutter SDK releases » [<https://fr.wikipedia.org/wiki/Flutter>] (consulté le 19 mai 2021) , [visité le 01/06/2021]

- [5] Site Internet de : [https://fr.wikipedia.org/wiki/Dart-\(langage\)](https://fr.wikipedia.org/wiki/Dart-(langage)) , [visité le 07/06/2021]

- [6] Site Internet de : Frederic Lardinois, « Microsoft Launches Visual Studio Code, A Free Cross-Platform Code Editor For OS X, Linux And Windows », TechCrunch, 29 avril 2015 (lire en ligne [archive]) , [visité le 05/06/2021]

- [7] Site Internet de : <https://fr.wikipedia.org/wiki/Firebase> , [visité le 07/06/2021]